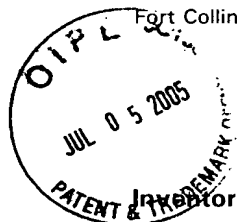


IN THE  
UNITED STATES PATENT AND TRADEMARK OFFICE



Inventor(s): Johnson, et al.

Confirmation No.: 7100

Application No.: 09/417,201

Examiner: Anya, Charles

Filing Date: 10-11-99

Group Art Unit: 2194

Title: System and Method for Intercepting, Instrumenting and Characterizing Usage of an Application Programming Interface

Mail Stop Appeal Brief-Patents  
Commissioner For Patents  
PO Box 1450  
Alexandria, VA 22313-1450

TRANSMITTAL OF APPEAL BRIEF

Sir:

Transmitted herewith is the Appeal Brief in this application with respect to the Notice of Appeal filed on 5-04-05.

The fee for filing this Appeal Brief is (37 CFR 1.17(c)) \$500.00.

(complete (a) or (b) as applicable)

The proceedings herein are for a patent application and the provisions of 37 CFR 1.136(a) apply.

( ) (a) Applicant petitions for an extension of time under 37 CFR 1.136 (fees: 37 CFR 1.17(a)-(d) for the total number of months checked below:

( ) one month	\$120.00
( ) two months	\$450.00
( ) three months	\$1020.00
( ) four months	\$1590.00

( ) The extension fee has already been filled in this application.

(X) (b) Applicant believes that no extension of time is required. However, this conditional petition is being made to provide for the possibility that applicant has inadvertently overlooked the need for a petition and fee for extension of time.

Please charge to Deposit Account 08-2025 the sum of \$500.00. At any time during the pendency of this application, please charge any fees required or credit any over payment to Deposit Account 08-2025 pursuant to 37 CFR 1.25. Additionally please charge any fees to Deposit Account 08-2025 under 37 CFR 1.16 through 1.21 inclusive, and any other sections in Title 37 of the Code of Federal Regulations that may regulate fees. A duplicate copy of this sheet is enclosed.

(X) I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, Alexandria, VA 22313-1450. Date of Deposit: 6-29-05

OR

( ) I hereby certify that this paper is being transmitted to the Patent and Trademark Office facsimile number \_\_\_\_\_ on \_\_\_\_\_

Number of pages:

Typed Name: Mary Meegan

Signature: Mary Meegan

Respectfully submitted,

Johnson, et al.

By David R. Risley

David R. Risley, Esq.

Attorney/Agent for Applicant(s)

Reg. No. 39,345

Date: 6-29-05

Telephone No.: (770) 933-9500



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re Application of:

Johnson, et al.

Serial No.: 09/417,201

Filed: October 11, 1999

Group Art Unit: 2194

Examiner: Anya, Charles

Docket No. 10982213-2

For: **System and Method for Intercepting, Instrumenting and Characterizing  
Usage of an Application Programming Interface**

**APPEAL BRIEF UNDER 37 C.F.R. §41.37**

Mail Stop: Appeal Brief-Patents  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, Virginia 22313-1450

Sir:

This Appeal Brief under 37 C.F.R. §41.37 is submitted in support of the Notice of Appeal filed May 4, 2005, responding to the Final Office Action mailed November 18, 2004.

It is not believed that extensions of time or fees are required to consider this Appeal Brief. However, in the event that additional extensions of time are necessary to allow consideration of this paper, such extensions are hereby petitioned under 37 C.F.R. §1.136(a), and any fees required therefor are hereby authorized to be charged to Deposit Account No. 08-2025.

07/06/2005 ZJU HAR1 00000022 082025 09417201

01 FC:1402 500.00 DA

### **I. Real Party in Interest**

The real party in interest is Hewlett-Packard Development Company, LP, a limited partnership established under the laws of the State of Texas and having a principal place of business at 20555 S.H. 249 Houston, TX 77070, U.S.A. (hereinafter "HPDC"). HPDC is a Texas limited partnership and is a wholly-owned affiliate of Hewlett-Packard Company, a Delaware Corporation, headquartered in Palo Alto, CA. The general or managing partner of HPDC is HPQ Holdings, LLC.

### **II. Related Appeals and Interferences**

There are no known related appeals or interferences that will affect or be affected by a decision in this Appeal.

### **III. Status of Claims**

Claims 1-5, 7-11, 13, 16, 17, and 21-26 stand finally rejected. No claims have been allowed. The final rejections of claims 1-5, 7-11, 13, 16, 17, and 21-26 are appealed.

### **IV. Status of Amendments**

This application was originally filed on October 11, 1999 with twenty (20) claims. In a Response filed December 19, 2002, Applicant amended claim 1 and canceled claim 6. In a Response filed May 9, 2003 along with a Request for Continued Prosecution Application (CPA), Applicant amended claims 2, 3, and 5. In a Response filed October 6, 2003, Applicant amended claims 7 and 13 and canceled claims 12, 19, and 20. In a Response filed February 13, 2001 along with a Request of Continued Examination (RCE), Applicant amended claims 1, 7, and 13. In a

Response filed July 1, 2004, Applicant amended claims 1, 3, 5, 7, 9-11, 13, 16, and 17, canceled claims 14, 15, and 18, and added new claims 21-26. In a Response filed January 5, 2005, Applicant again amended claim 1.

All of the above-identified amendments have been entered and no other amendments have been made to any of claims 1-5, 7-11, 13, 16, 17, and 21-26. The claims in the attached Claims Appendix (see below) reflect the present state of those claims:

#### **V. Summary of Claimed Subject Matter**

The claimed inventions are summarized below with reference numerals and references to the written description (“specification”) and drawings. All references are shown in the application at least where indicated herein.

Independent claim 1 recites a method for intercepting an event, the method comprising generating an event with an application program (90, Figs. 4A and 4C). Applicant’s specification, e.g., page 14, lines 19-20.

The method further comprises calling an application program interface (110, Figs. and 4A and 4C) to process the event. Applicant’s specification, e.g., page 14, lines 20-21.

The method further comprises receiving the event with the application program interface. Applicant’s specification, e.g., page 14, lines 21-22.

The method further comprises automatically determining without prompting from a user if an intercept library (130, Figs. 4A and 4C) is enabled to process the event. Applicant’s specification, e.g., page 14, lines 22-23.

The method further comprises, if the intercept library is enabled to process the event, automatically transmitting the event from the application program interface to a

generic interception communication interface (120, Figs. 4A and 4C) having at least one intercept event send handler (120, Fig. 4C), the generic interception communication interface maintaining communication between the application program interface and the intercept library. Applicant's specification, e.g., page 15, lines 3-6; page 14, lines 12-14.

The method further comprises transmitting the event from the generic interception communication interface to the intercept library with the at least one send handler. Applicant's specification, e.g., page 15, lines 3-6.

The method further comprises determining if the event is to be processed by the intercept library. Applicant's specification, e.g., page 15, lines 6-11.

Finally, the method of claim 1 comprises, if the event is to be processed by the intercept library, processing the event with the intercept library. Applicant's specification, e.g., page 15, lines 6-11.

Independent claim 7 recites an event interception system (100, Fig. 4A) for generic interception of events, comprising means for receiving and processing an event generated by an application program (90, Figs. 4A and 4B). Applicant's specification, e.g., page 14, lines 21-22.

The system further comprises means for determining whether any intercepting means is enabled to process the event. Applicant's specification, e.g., page 14, lines 22-23.

The system further comprises means for transmitting the event from the receiving and processing means to the intercepting means if the intercepting means is enabled to process the event. Applicant's specification, e.g., page 15, lines 3-6.

Finally, the system of claim 7 comprises intercepting means for receiving and processing the event. Applicant's specification, e.g., page 15, lines 3-6.

As is noted in claim 7, the means for transmitting the event comprises a generic interception communication interface (120, Fig. 4A and 4C) having at least one intercept event send handler (120, Fig. 4C), the generic interception communication interface maintaining communication between the intercepting means and the processing means. Applicant's specification, e.g., page 14, lines 12-14.

Independent claim 13 recites an event interception system (100, Fig. 4A) for generic interception of events, comprising an application program interface (110, Figs. 4A and 4C) that is configured to receive requests for service regarding events generated by an application program (90, Figs. 4A and 4B), the application program interface further being configured to determine if an intercept library is enabled to process the events and, if so, transmit the events to the intercept library. Applicant's specification, e.g., page 14, line 21 to page 15, line 6.

The system further comprises an intercept library (130, Figs. 4A and 4C) that is configured to process events. Applicant's specification, e.g., page 15, lines 12-20.

Finally, the system of claim 13 comprises a generic interception communication interface (120, Figs. 4A and 4C) that is configured to transmit events from the application program interface to the intercept library, the generic interception communication interface having at least one intercept event send handler for maintaining communication between the application program interface and the intercept library. Applicant's specification, e.g., page 14, lines 12-14; Fig. 4C, item 120.

## **VI. Grounds of Rejection to be Reviewed on Appeal**

The following ground of rejection is to be reviewed on appeal:

Claims 1-5, 7-11, 13, 16-17, and 21-26 have been rejected under 35 U.S.C. § 102(b) as being anticipated by Smale (U.S. Pat. No. 5,764,985). Applicant respectfully traverses this rejection.

## **VII. Arguments**

The Appellant respectfully submits that Applicant's claims are not anticipated under 35 U.S.C. § 102, and respectfully requests that the Board of Patent Appeals overturn the final rejections of those claims at least for the reasons discussed below.

Claims 1-5, 7-11, 13, 16-17, and 21-26 stand rejected under 35 U.S.C. § 102(b) as being anticipated by Smale (U.S. Pat. No. 5,764,985). Applicant respectfully traverses this rejection.

It is axiomatic that "[a]nticipation requires the disclosure in a single prior art reference of each element of the claim under consideration." *W. L. Gore & Associates, Inc. v. Garlock, Inc.*, 721 F.2d 1540, 1554, 220 USPQ 303, 313 (Fed. Cir. 1983). Therefore, every claimed feature of the claimed invention must be represented in the applied reference to constitute a proper rejection under 35 U.S.C. § 102(b).

In the present case, not every feature of the claimed invention is represented in the Smale reference. Applicant discusses the Smale reference and Applicant's claims in the following.

#### **A. The Smale Disclosure**

The Examiner primarily relies upon lines 10-31 and 41-67 of column 4 of the Smale reference in rejecting Applicant's claims. Those sections of the Smale reference provide as follows:

In the WOSA configuration, application programs 23, 24 initiate function calls via an application program interface (API) 25 requesting the performance of various operations. From the API 25, the call is received by a routing component 26, which routes the requested function call through a service provider interface (SPI) 27 to one or more appropriate service providers. One of the service providers, e.g. service provider 28, may handle the request directly, or alternatively may call on other, lower-level service providers (not shown). Another service provider 29 is provided to call on operating system functions when so required by the requesting function call. . . .

FIG. 2A illustrates the general flow of operation when a typical function call is made from an application program. As previously described, with a layered architecture, a request router 30 of the routing component 26 initially receives the requesting call from the API 25, as indicated by step (1) of FIG. 2A. In keeping with the invention, at step (2) the function call is subsequently passed from the request router 30 to the notification manager 32. In turn, the notification manager 32 sequentially notifies each of the extensions 34, 35, at steps (3) and (4), respectively, of the pending call before the call is made (pre-notification). This pre-notification step is also shown in FIG. 2B as step 16.

During this pre-notification procedure the extensions 34, 35 are called, enabling the extensions to execute their own procedures and perform operations according to their extended functionality thereby providing extensibility to the pending operation request. However, as described in more detail below, the extensions 34, 35 are limited in



how they can ultimately affect the subsequent passing of the pending request to the appropriate service provider.

Following pre-notification of the extensions, the notification manager returns control of the call to the request router 30 at step (5), which then routes the call to one or more appropriate service providers, such as the service provider 28, at step (6). The service provider 28 services the requesting call, and when completed, returns control to the request router 30 at step (7). The call or calls to the service providers are also shown in FIG. 2B as step 18. It should be noted that an extension can also be a service provider itself. Accordingly, service providers may register as extensions with the notification manager 32.

Applicant claims methods and systems that are distinct from those described in the foregoing excerpt.

#### **B. Applicant's Claims 1-5 and 21-22**

Applicant's independent claim 1 provides as follows (emphasis added):

1. A method for intercepting an event, the method comprising:

generating an event with an application program;

calling an application program interface to process the event;

receiving the event with the application program interface;

*automatically determining without prompting from a user if an intercept library is enabled to process the event;*

*if the intercept library is enabled to process the event, automatically transmitting the event from the application program interface to a generic interception communication interface having at least one intercept event send handler, the generic interception communication interface maintaining communication between the application program interface and the intercept library;*

*transmitting the event from the generic interception communication interface to the intercept library with the at least one send handler;*

determining if the event is to be processed by the intercept library; and

if the event is to be processed by the intercept library, processing the event with the intercept library.

Applicant discusses various limitations of the above claim in the following.

**1. Determining if an Intercept Library is Enabled to Process an Event**

As an initial matter, Smale does not teach “automatically determining without prompting from a user if an intercept library is enabled to process the event”. More particularly, no determination is made in the Smale system as to whether an intercept library is enabled to process the event. There is simply no disclosure from Smale on this point. Applicant further notes that, contrary to that alleged in the Final Office Action, column 4, lines 53-60 of the Smale reference does not teach this limitation. Instead, that portion of the Smale disclosure provides as follows (emphasis added):

*During this pre-notification procedure the extensions 34, 35 are called, enabling the extensions to execute their own procedures and perform operations according to their extended functionality thereby providing extensibility to the pending operation request. However, as described in more detail below, the extensions 34, 35 are limited in how they can ultimately affect the subsequent passing of the pending request to the appropriate service provider.*

Clearly, nothing in this excerpt discusses “determining” whether an intercept library is “enabled to process the event”. First, Smale’s cryptic reference to “pre-notification” does *not* equate to a teaching of “determining” whether an intercept library is “enabled to process” the event. Moreover, Applicant notes that, as is apparent from the above excerpt, it is Smale’s monitor extensions 34 and 35 that receive the “pre-notification.” Therefore, if, as is argued by the Examiner (see section “3” below), Smale’s service providers 28 and 29 are equivalent to Applicant’s claimed “intercept library”, *the pre-notification is not even provided to what the Examiner alleges to be Smale’s intercept library.*

Finally, in response to the Examiner’s comments contained in the Advisory Action, Applicant notes that column 8, lines 14-23 also fail to teach determining whether an intercept library is enabled to process an event. That portion of the Smale disclosure provides as follows:

After at least one extension has been registered and a valid function call from the API 24 is received at the notification manager 32, the notification manager 32 executes a preliminary procedure to prepare for the pre-notification of the appropriate types of extensions. It should be noted that invalid function calls from application programs (e.g., calls passing in bad parameters) are ordinarily detected at the MPR 26 and returned to the application program without ever reaching the notification manager.

First, this excerpt says nothing of making a “determination” as to whether an intercept library is “enabled”. Second, Smale is discussing the extensions 34 and 35, *not* the service providers 28 and 29 which the Examiner alleges to equate to Applicant’s intercept library (see section “3” below).

## 2. Transmitting an Event *If* the Intercept Library is Enabled

Smale further does not teach “if the intercept library is enabled to process the event, automatically transmitting the event from the application program interface to a generic interception communication interface having at least one intercept event send handler, the generic interception communication interface maintaining communication between the application program interface and the intercept library”. In particular, Smale does not teach making a determination as to whether an intercept library is enabled to process an event *before transmitting an event from an application program interface*. Again, Applicant notes that Smale’s mere reference to “pre-notification” does not equate to determining if an intercept library is enabled prior to transmitting an event to the intercept library. Moreover, as is noted above, it is Smale’s monitor extensions 34 and 35 that receive “pre-notification” and not Smale’s service providers 28 and 29, which the Examiner alleges to be Smale’s intercept library (see section “3” below).

As a further matter, Applicant notes that the claim limitation at issue explicitly requires: “if the intercept library is enabled to process the event, automatically transmitting the event *from the application program interface*”. Smale does not teach this. Instead, as is described above, Smale’s system transmits a call (“event”) from the API 25 to a request router 30, which transmits the call to the notification manager 32, *and then* the notification manager 32 provides the “pre-notification” to the monitor extensions 34 and 35. Smale, column 40-52. Therefore, Smale’s pre-notification occurs *well after* the call has been sent from the API. It logically follows then that Smale’s system does not make any determination as to the enablement of an intercept library and, only if the library is enabled, *then* transmits an event *from an API*.

### **3. “Intercept Library”**

Applicant further notes that Smale does not identify any “intercept library”. In the Final Office Action, it appears that the Examiner is arguing that Smale’s service providers 28 and 29 together comprise such an intercept library. See Final Office Action, page 4, lines 6-8. Applicant disagrees. *Nothing* in the Smale disclosure states that Smale’s service providers are libraries of any kind. Again, for a proper rejection under 35 U.S.C. § 102, the cited reference must disclose *each element* of the claim in question. Column 4, lines 61-67 and column 5, lines 1-4 of the Smale reference, which were identified in the Advisory Action, do not provide this missing teaching.

### **4. “Transmitting an Event from a Generic Interception Communication Interface to an Intercept Library**

Smale also does not teach or suggest “transmitting the event from the generic interception communication interface to the intercept library with the at least one send handler” for at least two reasons.

First, as is mentioned above, Smale does not disclose an “intercept library”.

Second, Smale does not disclose a “generic” interception communication interface. Although Smale describes a “routing component,” *nowhere* does Smale state that that component is generic within the meaning of Applicant’s disclosure. As is described in Applicant’s specification, the term “generic” refers to the ability of the interface to handle all API events without prior knowledge of the nature of those events. See, e.g., Applicant’s specification, page 7, lines 1-8.

In response to the Examiner’s point contained in the Advisory Action that “the specification should not be read into the claim,” Applicant agrees that additional

claim limitations contained in the specification should not be incorporated into the claims. However, Applicant notes that it is well established in the law that claim terms are to be *interpreted* in light of the specification. *Markman v. Westview Instruments, Inc.*, 52 F.3d 967, 34 USPQ2d 1321 (Fed. Cir. 1995)(in banc), *aff'd*, 517 U.S. 370, 38 USPQ2d 1461 (1996) (“Claims must be read in view of the specification, of which they are a part”).

## **5. Dependent Claims**

Applicant further notes that Applicant’s dependent claims contain limitations that are not taught by Smale. For example, in regard to claim 5, Smale does not teach “sending a message enabling the application program interface to process the event if the intercept library cannot process the event”. Again, as is noted above, Smale does not teach determining if an intercept library is enabled to process an event. It logically follows that Smale does not teach sending a message enabling the application program interface to process the event “if the intercept library cannot process the event”.

Regarding claim 21, Smale does not teach “invoking with the intercept library an event program that process the event”. Simply stated, Smale discloses no such “event program” that process events for an intercept library.

## **6. Conclusion**

For at least the foregoing reasons, Smale does not anticipate claim 1, or claims 2-5 and 21-22 which depend therefrom. Accordingly, claims 1-5 and 21-22 are allowable over Smale.

**C. Applicant's Claims 7-11 and 23-24**

Applicant's independent claim 7 provides as follows (emphasis added):

7. An event interception system for generic interception of events, comprising:

means for receiving and processing an event generated by an application program;

*means for determining whether any intercepting means is enabled to process the event;*

*means for transmitting the event from the receiving and processing means to the intercepting means if the intercepting means is enabled to process the event;* and

intercepting means for receiving and processing the event;

*wherein the means for transmitting the event comprises a generic interception communication interface having at least one intercept event send handler, the generic interception communication interface maintaining communication between the intercepting means and the processing means.*

Referring to independent claim 7, Smale does not teach or suggest “means for determining whether any intercepting means is enabled to process the event” or “means for transmitting the event from the receiving and processing means to the intercepting means if the intercepting means is enabled to process the event”. First, as is mentioned above in relation to claim 1, Smale does not describe determining whether interception means is enabled to process an event. Smale therefore fails to disclose means for performing that function. Second, Smale does not describe means for transmitting the event *if* the intercepting means is enabled to process the event.

As a further matter, Smale does not teach “wherein the means for transmitting the event comprises a generic interception communication interface having at least

one intercept event send handler, the generic interception communication interface maintaining communication between the intercepting means and the processing means” at least because Smale does not disclose a “generic” interception communication. Applicant refers to the discussion of claim 1 above.

For at least the foregoing reasons, Smale does not anticipate claim 7, or claims 8-11 and 23-24 which depend therefrom. Accordingly, claims 7-11 and 23-24 are allowable over Smale.

With specific regard to the claims that depend from claim 7, Smale does not teach “means for sending a message enabling the processing means to process the event if the intercepting means cannot process the event” as is provided in claim 11. Applicant refers to the discussion of claim 5 provided above.

Regarding claim 23, Smale does not teach “means for an event program that process the event” for reasons described above in relation to claim 21.

#### **D. Applicant’s Claims 13, 16-17, and 25-26**

Applicant’s independent claim 13 provides as follows (emphasis added):

13. An event interception system for generic interception of events, comprising:

an application program interface that is configured to receive requests for service regarding events generated by an application program, the application program interface further being configured *to determine if an intercept library is enabled to process the events and, if so, transmit the events to the intercept library*;

*an intercept library that is configured to process events*; and

*a generic interception communication interface* that is configured to transmit events from the application program interface to the intercept library, the generic interception communication interface



*having at least one intercept event send handler for maintaining communication between the application program interface and the intercept library.*

Regarding independent claim 13, Smale does not teach an application program interface that is configured to “determine if an intercept library is enabled to process the events and, if so, transmit the events to the intercept library”. As is noted above in relation to claim 1, Smale’s system is not described as comprising that functionality.

Smale further does not teach an “intercept library that is configured to process events”. Again, Smale does not disclose an intercept “library”.

Furthermore, Smale does not teach a “generic interception communication interface . . . having at least one intercept event send handler for maintaining communication between the application program interface and the intercept library”. As is stated in the foregoing, Smale does not disclose a “generic” interception communication interface.

For at least the foregoing reasons, Smale does not anticipate claim 13, or claims 16-17 and 25-26 which depend therefrom. Accordingly, claims 13, 16-17, and 25-26 are allowable over Smale.

With specific regard to dependent claim 17, Smale does not teach “wherein the intercept library is configured to send messages enabling the application programming interface to process events if the intercept library cannot process the events” for reasons described in relation to claim 5.

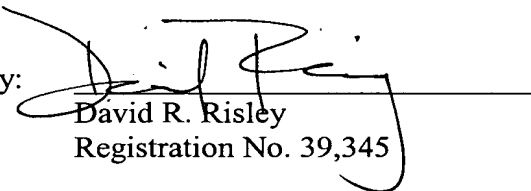
Regarding claim 25, Smale does not teach “wherein the intercept library is further configured to invoke an event program that processes events” for reasons described in relation to claim 21.

### **VIII. Conclusion**

In summary, it is Applicant's position that Applicant's claims are patentable over the applied prior art references and that the rejection of these claims should be withdrawn. Appellant therefore respectfully requests that the Board of Appeals overturn the Examiner's rejection and allow Applicant's pending claims.

Respectfully submitted,

By:

  
David R. Risley  
Registration No. 39,345

#### **Certificate of Mailing**

I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope, with sufficient postage, addressed to:  
Commissioner for Patents, P.O. Box 1450, Alexandria, VA,  
22313-1450 on:

6-29-05  
Mary M. Egan  
Signature

**Claims Appendix under 37 C.F.R. §41.37(c)(1)(viii)**

The following are the claims that are involved in this Appeal.

1. A method for intercepting an event, the method comprising:  
generating an event with an application program;  
calling an application program interface to process the event;  
receiving the event with the application program interface;  
automatically determining without prompting from a user if an intercept library is enabled to process the event;  
if the intercept library is enabled to process the event, automatically transmitting the event from the application program interface to a generic interception communication interface having at least one intercept event send handler, the generic interception communication interface maintaining communication between the application program interface and the intercept library;  
transmitting the event from the generic interception communication interface to the intercept library with the at least one send handler;  
determining if the event is to be processed by the intercept library; and  
if the event is to be processed by the intercept library, processing the event with the intercept library.
2. The method of claim 1, further including:  
defining a plurality of events to be intercepted.

3. The method of claim 2, wherein determining if the event is to be processed by the intercept library further includes:

finding the event to be processed in the plurality of events to be intercepted.

4. The method of claim 1, wherein the event is selected from the group consisting of function calls and operating system calls.

5. The method of claim 1, wherein processing the event includes:  
sending a message enabling the application program interface to process the event if the intercept library cannot process the event.

6. (Canceled)

7. An event interception system for generic interception of events, comprising:

means for receiving and processing an event generated by an application program;

means for determining whether any intercepting means is enabled to process the event;

means for transmitting the event from the receiving and processing means to the intercepting means if the intercepting means is enabled to process the event; and

intercepting means for receiving and processing the event;

wherein the means for transmitting the event comprises a generic interception communication interface having at least one intercept event send handler, the generic

interception communication interface maintaining communication between the intercepting means and the processing means.

8. The event interception system of claim 7, further comprising:  
means for defining a plurality of events to be intercepted.

9. The event interception system of claim 8, wherein the means for determining whether any intercepting means is enabled comprises:  
means for finding the event to be processed in the plurality of events to be intercepted.

10. The event interception system of claim 7, wherein the event is selected from the group consisting of function calls and operating system calls.

11. The event interception system of claim 7, further comprising:  
means for sending a message enabling the processing means to process the event if the intercepting means cannot process the event.

12. (Canceled)

13. An event interception system for generic interception of events, comprising:

an application program interface that is configured to receive requests for service regarding events generated by an application program, the application

program interface further being configured to determine if an intercept library is enabled to process the events and, if so, transmit the events to the intercept library;

an intercept library that is configured to process events; and

a generic interception communication interface that is configured to transmit events from the application program interface to the intercept library, the generic interception communication interface having at least one intercept event send handler for maintaining communication between the application program interface and the intercept library.

14-15. (Canceled)

16. The event interception system of claim 13, wherein the events are selected from the group consisting of function calls and operating system calls.

17. The event interception system of claim 13, wherein the intercept library is configured to send messages enabling the application programming interface to process events if the intercept library cannot process the events.

18-20. (Canceled)

21. The method of claim 1, wherein processing the event comprises invoking with the intercept library an event program that processes the event.

22. The method of claim 1, further comprising returning an output from the intercept library to the application program interface for transmission to the application program.

23. The event interception system of claim 7, wherein the intercepting means comprises means for invoking an event program that processes events.

24. The event interception system of claim 7, further comprising means for returning an output from the intercepting means to the means for receiving and processing an event.

25. The event interception system of claim 13, wherein the intercept library is further configured to invoke an event program that processes events.

26. The event interception system of claim 13, wherein the intercept library is further configured to return an output to the application program interface.

**Evidence Appendix under 37 C.F.R. §41.37(c)(1)(ix)**

There is no extrinsic evidence to be considered in this Appeal. Therefore, no evidence is presented in this Appendix.



**Related Proceedings Appendix under 37 C.F.R. §41.37(c)(1)(x)**

There are no related proceedings to be considered in this Appeal. Therefore,  
no such proceedings are identified in this Appendix.